

Pia Virtanen

# Vaihtuvat sisältöelementit mainosvideoissa

Metropolia Ammattikorkeakoulu

Insinööri (AMK)

Mediatekniikan koulutusohjelma

Insinööriytyö

18.6.2018

|   |  |
|---|--|
| Tekijä<br>Otsikko   | Pia Virtanen<br>Vaihtuvat sisältöelementit mainosvideoissa |
| Sivumäärä<br>Aika   | 44 sivua + 1 liite<br>18.6.2018                            |
| Tutkinto  | Insinööri (AMK)  |
| Koulutusohjelma   | Mediatekniikka   |
| Suuntautumisvaihtoehto  | Digitaalinen tekniikka                                     |
| Ohjaaja   | Lehtori Jonna Eriksson                                     |
| <p>Insinöörityön tarkoituksena oli perehtyä, miten käytetään vaihtuvia sisältöelementtejä mainosvideoissa. Mainonnassa lisääntyneen personoinnin käyttö antoi idean tutkia, miten personointia voidaan käyttää yhä yleistyvässä videomainonnassa mahdollisimman helposti ilman aiemmin käytettyjä kolmansien osapuolten liitännäisiä ja ilman suurta koodaustaitoa.</p> <p>Aihetta lähestyttiin ensin painetussa mediassa käytössä olevien personointitapojen kautta, ja personoinnin videomainonnan tekniikkana päätettiin tutkia, miten nopeasti yleistynyt ja usean vuoden käytössä ollut HTML5-tekniikka sopii siihen.</p> <p>Työssä tutkittiin teoriassa ja testattiin käytännössä, miten HTML5:n uudet mediaelementit, canvas ja video, soveltuvat videon personointiin. Tutkimuksessa käytettiin lähteinä ja apuna sekä kirjallisuutta ja internetsivustoja että YouTube-videoita. Niiden avulla päästiin kokeilemaan, ensin valmiiden mallien myötä, miten helposti voitiin videota pikselimanipuloida. Myöhemmin kuvattiin oma video ohi ajavasta pakettiautosta, jonka kylkeen lisättiin canvas-elementin avulla muuttuvan tiedon osio. Tähän osioon tehtiin testattavaksi kaksi erilaista kuvaa, toinen JPG-bittikartakuva ja toinen läpinäkyvyyttä tukeva PNG-kuva.</p> <p>Työssä näytettiin, että peruskoodaustaidoilla oli mahdollista tehdä lyhyellä koodilla videon personoitu osio, jota testattiin lisäämällä siihen kuva kuvapankista. Kuvapankin käyttö personointiin todettiin toimivaksi tavaksi, sillä siinä oli helppo muokata kuvaa ja jopa vaihtaa se, ilman että alkuperäiseen koodiin koskettiin.</p> <p>Työssä todettiin, että HTML5 on hyvä tekniikka personointiin, mutta täysin ammattimaisia videopersonointeja varten on vieläkin käytettävä valmiiksi tehtyjä ohjelmia, sillä HTML5:n puutteena oli se, että canvas-elementin liikkuminen pitää laskea käsin.</p> <p>Insinöörityötä voidaan hyödyntää, jos halutaan tietoa, miten mainonnan personointi on kehittynyt ja halutaan tutustua HTML5:n käyttö- ja käsittelymahdollisuuksiin videoissa.</p> |  |
| Avainsanat  | HTML5, canvas, muuttuva tieto, personointi                 |

|   |   |
|---|---|
| Author<br>Title   | Pia Virtanen<br>Variable content elements in advertisement videos |
| Number of Pages<br>Date   | 44 pages + 1 appendix<br>18 June 2018                             |
| Degree  | Bachelor on Engineering   |
| Degree Programme  | Media Technology  |
| Specialisation option   | Digital Technology  |
| Instructor  | Jonna Eriksson, Senior Lecturer                                   |
| <p>This final study examines dynamic content elements in advertisement videos. The increased use of personalization in advertising gave the idea to investigate how personalization can be used in videos, as the use of videos in marketing is increasing. It was examined how this could be implemented without third-party plugins and without advanced computing knowledge.</p> <p>The subject was first approached by examining existing techniques for personalizing print media. The tools chosen for personalized videos was basic HTML5 and JavaScript to see how well that would suit the purpose.</p> <p>Investigation was conducted in theory, and in practice it was tested how well the new media elements of HTML5, canvas and video, were suitable for the personalization of videos. Sources and help for the study were taken from literature, internet pages and YouTube-videos.</p> <p>With these, it was possible to test, first through ready-made models, how easy pixel manipulation was. Later on, a video of a truck driving by was recorded. On the side of the truck, with the help of a canvas element, a dynamic overlay element was added. For this element two different images were made, one JPG-bitmap image and another transparency supported PNG-image.</p> <p>In the work it was presented how, with basic computing skills, it was possible to create a personalized section, with a short code. The section was then filled with an image that had been uploaded into an image bank. The personalization functionality was confirmed by manipulating and changing the image from within the image bank without altering the original source code.</p> <p>The study concluded that there is adequate techniques available in HTML5 for personalization, but for professional video personalization it is still more convenient to utilize commercially software, mainly because in HTML5 one will have to calculate the movement of the canvas overlay element by hand.</p> |   |
| Keywords  | HTML5, canvas, variable data, personalization                     |

## Sisällys

### Lyhenteet

|     |  |    |
|-----|--|----|
| 1   | Johdanto   | 1  |
| 2   | Liikkuvan kuvan personointitarve markkinoinnissa             | 3  |
| 2.1 | Vaihtuvan tiedon ja personoinnin historiaa                   | 3  |
| 2.2 | Personoidun markkinointimateriaalin tuottaminen              | 4  |
| 2.3 | Web-to-print ja muuttuvan tiedon tulostaminen                | 6  |
| 2.4 | Muuttuvan tiedon tulostaminen                                | 6  |
| 2.5 | Muuttuvan tiedon käyttö taitossa                             | 7  |
| 2.6 | Muuttuvan tiedon käyttö mainoksissa ja internetissä          | 9  |
| 3   | Vaihtuva data liikkuvana elementtinä videossa                | 9  |
| 4   | HTML5:n uudet elementit: <canvas> ja <video>                 | 11 |
| 4.1 | HTML5 <canvas> -elementti                                    | 11 |
| 4.2 | HTML5 <video> -elementti                                     | 14 |
| 5   | HTML5 video + canvas ja pikselimanipulaatio                  | 18 |
| 6   | Testityö ja työn vaiheet                                     | 22 |
| 6.1 | HTML-sivun perusrakenne                                      | 23 |
| 6.2 | Videon kuvaaminen ja HTML-tiedostoon lisääminen              | 24 |
| 6.3 | Sivun lataaminen ennen toimintoja ja piirtoalueen lisääminen | 25 |
| 6.4 | Olion tekeminen  | 25 |
| 6.5 | Olion piirtäminen ja lataaminen                              | 26 |
| 6.6 | Aikavälit ja piirtoalueen tyhjennys                          | 26 |
| 6.7 | Liikkuva tuotokuva videossa                                  | 27 |
| 7   | Tekniikan kehittyminen liikkuvan kuvan muokkaukseen          | 31 |
| 7.1 | HTML5 yleisstandardina ja sen tulevaisuus                    | 31 |
| 7.2 | Edut ja mahdollisuudet                                       | 33 |
| 8   | Tulokset ja johtopäätös                                      | 33 |
|     | Lähteet  | 35 |
|     | Liitteet   |    |
|     | Liite 1. Testityön koodi                                     |    |

## Lyhenteet

|        |  |
|--------|--|
| 2D     | two-dimensional, kaksiulotteinen                                       |
| 3D     | three-dimensional, kolmiulotteinen                                     |
| API    | Application Programming Interface, ohjelmointirajapinta                |
| CDN    | Content Distribution Network   |
| CSS    | Cascading Style Sheets   |
| CSS3   | Cascading Style Sheets versio 3  |
| DAM    | Digital Asset Management   |
| DTP    | Desktop Publishing   |
| FPS    | Frames per Second, kuvataajuus   |
| HTML   | HyperText Markup Language  |
| HTML5  | HyperText Markup Language versio 5                                     |
| JPG    | Joint Photographic Experts Group, häviöllinen bittikarttatiedostomuoto |
| JSON   | JavaScript Object Notation   |
| MySQL  | Structured Query Language  |
| ODBC   | Open Database Connectivity   |
| PDF/VT | Portable Document Format / variable data and transactional             |
| PIM    | Product Information Management   |
| PNG    | Portable Network Graphics, häviötön bittikarttatiedostomuoto           |
| POS    | Point of Sale  |
| REST   | Representational State Transfer  |
| SOAP   | Simple Object Access Protocol  |
| URL    | Uniform Resource Locator   |
| VDP    | Variable Data Printing   |
| VP8    | Videokoodekki  |
| VP9    | Videokoodekki  |
| W3C    | World Wide Web Consortium  |
| WebGL  | Web Graphics Library   |
| WWW    | World Wide Web   |
| XML    | Extensible Markup Language   |

## 1 Johdanto

On jo pitkään tiedetty, että personoitu markkinointi toimii paremmin kuin massamarkkinointi. Digitaalisten painokoneiden tulon myötä 1990-luvulla digitaalinen painaminen mullisti painetun markkinoinnin, sillä uusi painoteknologia teki mahdolliseksi kustannustehokkaan personoidun markkinointimateriaalin tuottamisen helposti ja nopeasti. Myöhemmin kehitetyillä uusilla teknologioilla ja ohjelmistoilla voitiin tehdä muun muassa tietokantataittoa, jonka avulla pystyttiin osittain automatisoimaan tuoteluetteloiden tuotantoprosessi.

Internet-markkinointi on entistä enemmän suuntautumassa kohti personoitua markkinointia. Internetissä käyttäjä ja hänen kiinnostuksenkohteensa tunnistetaan ja usein myös käyttäjän sijainti tiedetään, jolloin voidaan tehdä hyvinkin kohdistettua markkinointia.

Liikkuvan kuvan käyttö markkinoinnissa on lisääntynyt räjähdysmäisesti, minkä seurauksena voidaan nähdä, että tulevaisuuden trendinä tarvitaan helppo tapa tuottaa personoituja markkinointivideoita. Tämä valittiin insinööriyön aiheeksi, ja työssä haluttiin tutkia, kuinka helposti tällä hetkellä voidaan tehdä personoituja videoita käyttämällä yleisiä työkaluja, ilman kolmannen osapuolen ohjelmia, pelkän koodin avulla ilman suurta ohjelmointitaitoa.

Insinööriyössä testataan, miten voidaan helposti lisätä videoon muuttuvan datan osio, joka liikkuu mahdollisimman saumattomasti videon mukana. Samoin selvitetään, miten yleisillä työkalulla voidaan luoda videoon liikkuvia elementtejä ja miten voidaan tuottaa muuttuvaa sisältöä näihin elementteihin. Näiden vaihtuvien sisältöelementtien tekniikkana tutkitaan jo joitakin vuosia käytössä ollutta HTML5-tekniikkaa ja sen canvas-elementtiin upotettavaa videoelementtiä. HTML5 on nykyisin yleisesti käytössä ja hyvin tuettu eri selaimissa. Lisäksi sillä on monipuolisia mahdollisuuksia sisällöntekemiseen, ja tämän takia sen käyttö lisääntyy jatkuvasti.

Työn alussa taustana esitellään personoidun ja vaihtuvan datan sekä tiedon käyttöä painetussa mediassa ja sitä, miten tarve on tekniikan, laitteiden ja internetin kehittymisen myötä johtanut lisääntyvään liikkuvan kuvan suosioon mainonnassa ja markkinoinnissa. Työssä käydään läpi tarvittavat tekniikat ja elementit, ja lopussa tutkitaan tekniikan automaatiomahdollisuutta hakea ja vaihtaa sisältöä ”lennossa” kuvapankista tai tietokannasta. Lyhyesti käydään myös läpi tekniikan integraatiomahdollisuudet. Lopuksi selvitetään, miten tavoitteissa onnistuttiin.

## 2 Liikkuvan kuvan personointitarve markkinoinnissa

Kuvassa 1 esitetään mediamainonnan osuus Suomessa vuonna 2016, ja siinä näkee, että vuodessa verkkomedia on kasvanut 3 prosenttiyksikköä 28 prosenttiin. Vuonna 2016 koko mediamainontaan käytettiin 1 168 miljoonaa euroa, joten mediamainonnassa ei puhuta pienistä rahasummista. Verkkomedian kasvu selittyy yhä enenevässä määrin kasvaneesta mobiililaitteiden käytöstä ja niissä käytetystä mobiilimainonnasta. (17.)



Kuva 1. Mediamainonta Suomessa vuonna 2016 (17).

### 2.1 Vaihtuvan tiedon ja personoinnin historiaa

Vaihtuvaa dataa on käytetty ennen digitaalista painamistekniikkaakin, esimerkiksi lentolippujen tulostamisessa ja nimi- ja osoitetietojen tulostamisessa erilaisissa transaktiotuotteissa ja suoramarkkinointikirjeissä. Ennen digitaalista painotekniikkaa on käytetty mustesuihkumenetelmää. (45, s. 9.)

Erilaisia painettuja suorakirjeitä, pakkauksia ja liikekirjeitä, kuten laskuja, postitetaan joka päivä suuria määriä, ja yritykset käyttävät siihen paljon rahaa. Suuret painosmäärät ja vaatimukset nopeista läpimenoajoista ovat nopeuttaneet laitteiden, kuten korkealaatuisten tulostimien, kehitystä. Myös ohjelmat ovat kehittyneet, ja esimerkiksi

muuttuvan tiedon tulostukseen (Variable Data Printing, VDP) on kehitetty ohjelmistoja, jotta saataisiin alennettua yksittäisen tulostetun tuotteen kustannuksia. (16.)

Toimistolaitteita valmistava yhdysvaltalainen Xerox-yhtiö julkaisi vuonna 1995 InterDocin, joka oli ensimmäinen asiakas-palvelinohjelma, jolla voitiin tehdä tarvepainatusta (print-on-demand) internetin avulla senaikaisilla käytössä olleilla tulostimilla. Xeroxin kehittämä InterDoc perustui avoimeen ohjelmistoratkaisuun. (18.)

Graafisessa tekniikassa tietokoneistuminen sekä työasemataiton ja digitaalisen painokoneen käyttäminen teki vaihtuvan datan käytön mahdolliseksi ja helpoksi, ja sillä oli muitakin etuja kuin personoidun painotuotteen tekeminen. Tekniikan helppokäyttöisyys ja säästöt pienempien painosmäärien painokustannuksissa olivat iso etu yrityksille. (11.)

Vuonna 2008 Adobe julkaisi PDF-tiedostojen tuottamiseen tarkoitetun Adobe PDF Print Engine 2 -tulostusmoottorin (APPE2), jotta muuttuvan tiedon tulostaminen olisi helpompaa. APPE-tulostusmoottoria käytetään prosessoimaan PDF-VT-formaattia (Portable Document Format / variable data and transactional) personointityönkuluissa. Nykyään APPEsta on käytössä versio 4, APPE4. Vuodesta 2010 PDF/VT on ollut ISO-standardoitu muuttuvan tiedon (VDP) tulostukseen. (14; 39.)

## 2.2 Personoidun markkinointimateriaalin tuottaminen

Suomessa ensimmäinen digitaalinen painokone otettiin käyttöön vuonna 1995 (15, s. 92). Digitaalinen painopinnanvalmistus oli otettu käyttöön jo aiemmin, mutta digitaalinen painotekniikka yleistyi vasta 2000-luvun vaihteessa ja mullisti samalla olemassa olevan painotekniikan uudella mahdollisuudella käyttää painotuotteissa personoitavaa tietoa. Nyt pystyttiin tuottamaan yksilöllistä markkinointimateriaalia kustannustehokkaasti, mitä ei voitu tehdä aiemmin vanhoilla painotekniikoilla. Digitaalisen painotekniikan myötä alettiin puhua tarvepainatuksesta, Print on Demand -painatuksesta. (8; 9, s. 10–11 ja s. 19–20; 10.)

Digitaalisessa painotavassa painotuotteiden personointi on mahdollista, koska jokaisen uuden painoarkin painopinnalle muodostetaan painettava kuva uudelleen, jolloin sen ei tarvitse olla samanlainen kuin edellisen painopinnan kuva. Näin voidaan personoida jokaiseen painoarkkiin eri nimi, osoite, muu teksti, kuva tai kokonainen sivu. (8; 9, s. 13; 10.)

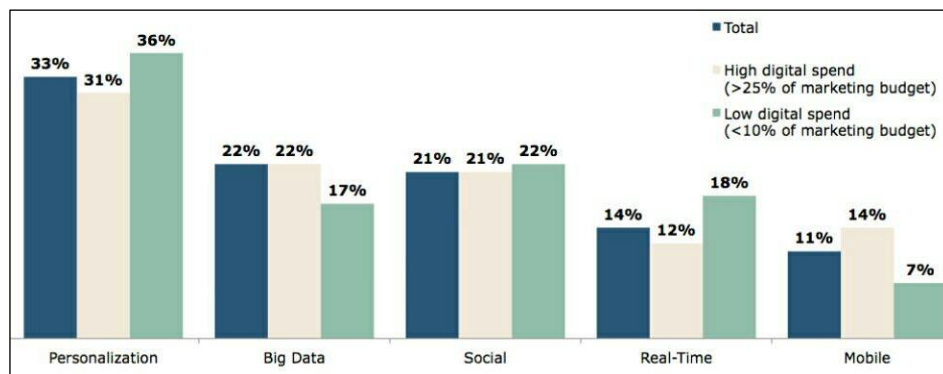
Personoitujen digitaalisten painotuotteiden vaihtuva tieto voidaan jakaa eri tyyppisiin: se voi olla tekstitietoa, kuva tai taittopohja. Vaihtuvan tiedon tyyppinä puhutaan personoinnista, räätälöinnistä ja varioinnista. Vaihtuvat elementit määritellään tulostettaviin dokumentteihin, joihin vaihtuva tieto sijoitetaan sivukohtaisesti. Ennen sivujen tulostusta vaihtuva tieto haetaan tietokannasta. (15, s. 102.)

Tekniikan kehityksen myötä markkinointikin on osittain automatisoitunut, mutta se ei tarkoita, etteikö ihminen edelleen päättäisi, mitä ja miten markkinoidaan. Markkinointiviesteihin halutaan personointia, ja siihen tarvittavaa tietoa voidaan kerätä vaikkapa käyttäjän toimintatavoista ja näin optimoida tarkasti, mitä käyttäjälle näytetään. Digitaaliseen markkinointiin kuuluu yleensä neljää eri osaa: tarvitaan markkinointiviesti, -kanava, internetsivusto ja tulosten mittaaminen. (5.)

Personoitu markkinointi on yleisesti käytetty tapa kohdistetuissa sähköposti-markkinointikirjeissä, erilaisissa kampanjoissa ja myös painetuissa suoramainontakirjeissä (3; 4). Useissa tutkimuksissa on osoitettu, että ihmiset reagoivat jopa 60 % enemmän personoituun mainosmateriaaliin kuin personoimattomaan mainokseen (11; 36). Kuitenkin tekniikan käyttöönotto koetaan haastavaksi (IT-esteet, vanhat käytössä olevat järjestelmät sekä budjettirajoitukset ja resurssipula), vaikka markkinointipäätäjät tiedostavat personoinnin merkityksen kasvulle (37).

Adoben yritysten markkinointipäätäjille teettämästä kyselystä kuvassa 2 näkyy, että vuonna 2014 yhdysvaltalaiset markkinointipäätäjät ovat vastanneet personoinnin

yhtiönsä markkinoinnin tärkeimmäksi kohdealueeksi tulevaisuudessa, riippumatta siitä, kuinka paljon rahaa markkinointiin on käytössä.



Kuva 2. Mainostajien tärkeimmät markkinointikohteet vuonna 2014 Yhdysvalloissa (36).

### 2.3 Web-to-print ja muuttuvan tiedon tulostaminen

Ensimmäisen kerran web-to-print-termiä käytti Jim Frew vuonna 1999 online-artikkelissaan (12). Kaupalliseen käyttöön tarkoitetuissa web-to-print-ohjelmissa voi olla mukana sekä tarvepainatusmateriaalia (print-on-demand) että varastoon valmiiksi painettua materiaalia. Nämä print-on-demand-dokumentit voivat sisältää joko staattista sisältöä tai muuttuvan tiedon (VDP) elementtiosioita. Usein käytetty tapa on personoida markkinointimateriaalin tuote- tai asiakastiedot tietokannasta. VDP on suuntautumassa kohti massaräätälöintiä (mass customization), kun taas web-to-print keskittyy enemmän muutosten tekoon jokaiseen tilaukseen erikseen. (12; 14 ja 45.)

### 2.4 Muuttuvan tiedon tulostaminen

Kun konfiguroidaan metadataa ja sen avulla ohjataan tulostuksen työprosessia, PDF/VT (Portable Document Format / variable data and transactional) on joustava työkalu siihen tarkoitukseen. Massapostitukset on tarkoituksenmukaista pystyä ryhmittelemään esimerkiksi vastaanottajan, postinumeron tai sukupuolen mukaan kuoritusta tai lähetystä varten. Kun tulostuksessa käytetään muuttuvaa tietoa, muuttuvan tiedon aikaansaamat muutokset pitää ottaa huomioon kokonaissivumäärissä, tekstikappaleissa ja muissa muuttuvissa graafisissa elementeissä. Yksi PDF/VT:n vahvuus on tuki striimatulle

tietovirralle, eli tietoa voidaan lukea sitä mukaa, kun se saapuu vastaanottajalle, eikä tarvitse odottaa koko PDF-tiedoston latautumista vastaanottajan tietokoneelle.

Erilaisia VDP-ohjelmia on käytössä monissa julkaisuyhteisissä: liikekirjeiden vastaanottajan nimi- ja osoitetiedoissa ja yksilömarkkinoinnissa, jossa graafiset elementit ja vaihtuvan tekstin komponentit on valittu jokaiselle vastaanottajalle. Erilaiset yhdistelmät voidaan näin määritellä vastaanottajan kiinnostuksenkohteiden tai aiemman ostokäyttäytymisen mukaan. (16.)

## 2.5 Muuttuvan tiedon käyttö taitossa

### Tietokantataitto

Työasemataiton (Desktop Publishing, DTP) ohjelmistoja, lähinnä Adobe Indesignia, käytetään painotuotteiden sivujen taittamiseen. Nykyään on mahdollista taittaa sivustot hakemalla sisältöä ulkopuolisesta lähteestä, kuten erilaisista tietokannoista ja Excel- taulukoista tai kutsumalla URL-linkkejä kuvien hakemiseen esimerkiksi kuvapankista. Tällaista taittotapaa kutsutaan tietokantataitoksi.

Tietokantataitto auttaa taittajan työtä nopeuttamalla sivujen taittamista, ja sitä voidaan käyttää yleisessä graafisen alan taitto-ohjelmassa Indesignissa. Tietokantataitossa tehdään tavallaan automaattinen esitaitto kuvien ja tekstin suhteen, minkä jälkeen taittaja viimeistelee taiton. Tietokantana voidaan käyttää joko tekstitiedostoa tai jo olemassa olevaa XML- tai ODBC-tietokantaa. Tietokannan tiedot, sekä tekstit että kuvat, ovat Indesignissa edelleen linkitettyinä käytettyyn tietokantaan. Tietokantaan voidaan tehdä tarvittaessa muutoksia, ja taitto pysyy linkkien ansiosta ajantasalla. (1.)

### Tuotekuvat ja -informaatio

Adobe Indesign -taitto-ohjelmaan voidaan lisätä esimerkiksi EasyCatalog-liitännäinen, jolla taitto voidaan automatisoida niin, että sisältöä haetaan suoraan ulkopuolisista lähteistä. Lähteinä voidaan käyttää esimerkiksi ODBC-tietokantaa, tuotetiedonhallinta-

järjestelmää (PIM), digitaalista aineistohallintajärjestelmää (DAM) tai Excel-taulukkoa. Tietokannan sisältö voidaan julkaista suoraan Indesigniin, niin että jokainen kuva ja teksti taitetaan juuri oikeille sivuille, oikeisiin paikkoihin. Kun ulkopuolisen lähteen sisältö muuttuu, päivittyy myös taitto automaattisesti. Tämä nopeuttaa ja tehostaa tuotantoprosessia huomattavasti, koska taittaja voi keskittyä vain taiton hienosäätöön ja nopeuttaa näin taitettavan luettelon koko kommentointi- ja hyväksyntäprosessia. (1; 13.)

## Tuotetiedonhallinta

Tuotetiedonhallinnalla (Product Information Management, PIM) tarkoitetaan niitä prosesseja ja teknologioita, jotka keskittyvät hallinnoimaan keskitetysti tietoa tuotteista. Tämä kohdistuu erityisesti tietoihin, joita tarvitaan tuotteiden markkinointia ja myyntiä varten, joko yhden tai useamman julkaisukanavan kautta. (19, s. 3.)

Tuotetiedonhallintaa käytetään esittämään yhdessä paikassa oleva tieto tuotteesta, ja näin vältetään siltä, että tuotteesta löytyisi erilaista, vaikkapa vanhentunutta tietoa, jota vahingossa voitaisiin käyttää. Tuotetiedonhallinta kerää tietoa eri lähteistä, tallentaa tiedot kerran ja julkaisee ne erilaisiin valittuihin kanaviin. Tuotetiedonhallintajärjestelmän avulla voidaan helpommin kohdistaa, jakaa ja ylläpitää tuotetietoa niin, että otetaan huomioon eri julkaisukanavien vaatimukset, kielistykset tai ajankohdat. Kytkemällä PIM esimerkiksi videoon on mahdollista hakea URL-linkin kautta kuvapankista vaihtuvaa tietoa videoon. (19, s. 3; 13.)

## Kuvapankki ja hajautettu sisällönjakelu

Digitaalisen aineistohallinnan (Digital Asset Management, DAM) avulla voidaan ylläpitää digitaalista sisältöä jatkokäyttöä varten. Sen avulla tallennetaan, ylläpidetään ja jaetaan esimerkiksi tuotekuvia ja niiden metatietoja jatkokäyttöä varten muun muassa tuoteluetteloihin, www-sivustoihin, verkkokaupoihin tai POS:iin (Point of Sale, myymälämateriaali). Rajapintojen avulla (esim. REST tai SOAP) voidaan hakea kuvat toiseen järjestelmään, esimerkiksi aiemmin kuvattuun Indesign-taittoon. On myös mahdollista julkaista tuotekuvia ja -tiedostoja käyttämällä hajautettua sisällönjakeluverkostoa (Content Distribution Network, CDN), jossa kuvat jaetaan

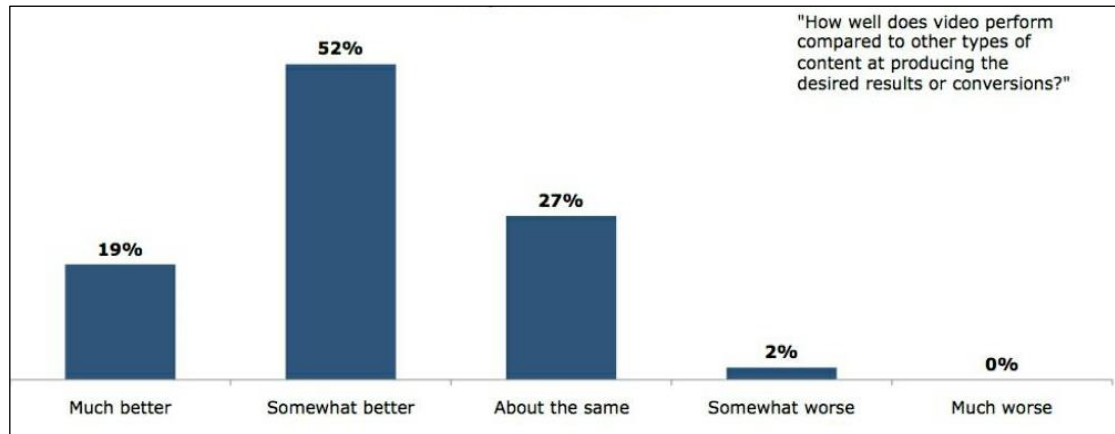
käyttäjille lähimmältä mahdolliselta palvelimelta, jonka välimuistiin ne on tallennettu. Hajautettu sisällönjakeluverkosto nopeuttaa sivustojen käyttöä ja on hyödyllinen sellaisille yrityksille, joilla on käyttäjiä monista eri maista ja joiden sivustot sisältävät paljon kuvia ja tiedostoja. Tuotekuvia ja -tiedostoja ei tällöin tarvitse tallentaa erikseen esimerkiksi www-sivuja tuottavaan julkaisujärjestelmään. (13; 41.)

## 2.6 Muuttuvan tiedon käyttö mainoksissa ja internetissä

Älypuhelinien ja tablettien suosion kasvu on lisännyt kiinnostusta personoituun mainontaan. Muun muassa HTML5:n paikannustekniikka auttaa siinä, ja sijaintitietoja myös myydään mainostajille, jotka näin voivat kohdistaa mainontaansa paremmin. Personointi ei rajoitu pelkkään internetmainontaan, vaan siitä saadun käyttäjätiedon perusteella tavalliset televisiomainoksetkin voivat olla erilaisia ja eripituisia kuin naapurilla tai digitaalinen ulkomainos näyttää toiselle ihmiselle eri mainoksen kuin takana kävelijälle. Tämä liittyy ohjelmalliseen ostamiseen, automatiikkaan, joka myy erilaiset digimainospaikat sekunnin murto-osissa käyttäjistä kerätyn tiedon perusteella. Käyttäjistä kerätystä tiedosta sivustot alkavat kertoa aiempaa paremmin, sillä keväällä 2018 otetaan käyttöön EU:n tietosuoja-asetus, jolla käyttäjä voi hallinnoida hänestä kerättyä tietoa ja päättää, kenelle sitä luovutetaan (50).

## 3 Vaihtuva data liikkuvana elementtinä videossa

Videoiden käyttö markkinointitarkoituksiin on lisääntynyt nopeasti. Markkinoijat uskovat sen käytön tulevan yhä tärkeämmäksi osaksi markkinointi- ja myyntiprosesseja ja ovat valmiita investoimaan videosisältöön (video content) ja vaativat tekniikalta myös tuloksia (kuva 3). Silti vain osa markkinoijista on integroinut järjestelmän osaksi CRM- tai markkinointityönkulkuaan. (38.)



Kuva 3. Videosisällön markkinoititehokkuus vuonna 2014 B2B-markkinoijien mukaan. (49).

Internetin kehittymisen myötä suuntaus käyttää personoitua markkinointia on vain voimistunut, kun on tullut mahdolliseksi kerätä helposti käyttäjäkohtaista tietoa tietovarastoihin (Data Warehouse). Mahdollisuus analysoida kuluttajien käyttäytymistä on auttanut yrityksiä kohdistamaan oikea markkinoitisisältö oikeille kohderyhmille.

Vuonna 2014 Adoben teettämän tutkimuksen mukaan (Digital Roadblock: Marketers Struggle to Reinvent Themselves), markkinoijat olivat sitä mieltä, että markkinoinnin personointi on tärkeintä markkinoinnin edistymiselle (61).

Video ja liikkuva kuva on tehokas markkinoitikanava. Videoiden teko voidaan tuottaa kustannustehokkaammin, jos samaa videopohjaa voidaan käyttää uudelleen vaihtamalla vain osa sisällöstä, käyttämällä muuttuvaa sisältöä.

Nykyisin, kun markkinoinnissa puhutaan personoinnista, ei tarkoiteta vain yhden tuotteen personointimahdollisuutta, vaan samalla pitää pystyä personoimaan tuotteet kaikissa markkinoitikanavissa (62). Sekä videot että niiden personointi lisäävät ostotapahtumien kasvua, ja yhdistämällä ne molemmat saadaan aikaan tehokkaampaa markkinointia (63).

Jo nyt on käytössä paljon videoita, jotka perustuvat videopohjiin, joihin vain liitetään vaihtuvaa dataa. Vaihtuva data on yleensä tekstiä ja/tai kuvaa sijoitettuna videossa tarkkaan määriteltyyn paikkaan.

Liittämällä vaihtuva data videoon niin, että katsoja kokee sen olevan saumattomasti osa taustalla olevaa videota, saadaan videosta entistä personoidumman oloinen. (13.)

Videon katselminen selaimessa

Aiemmin videon katselamiseen selaimessa tarvittiin erillinen liitännäinen (plug-in). Useimmiten käytössä oli Adoben Flash Player (alun perin Macromedian omistama ennen Adobelle siirtymistä vuonna 2005). Flash Playeriä käytettiin internetselaimiin asennettavan liitännäisen avulla. Sitä voitiin käyttää myös sen sisältämän ActionScript-ohjelmointikielen avulla. Vuonna 2017 Adobe ilmoitti lopettavansa Flashin kehittämisen. (64.)

Myös Microsoftilla on Adobe Flash Playerin tapainen selaimen asennettava liitännäinen, Silverlight, jolla voidaan toistaa videoita. Microsoft julkaisi Silverlightin vuonna 2007, ja sen kehittäminen lopetettiin vuonna 2013. HTML5 julkaistiin virallisesti 28. lokakuuta 2014, jolloin siitä tuli World Wide Web Consortiumin (W3C) suositus. (25; 65.)

Flash Player on ollut näistä kahdesta liitännäisestä käytetympi: vuonna 2011 tammikuussa verkkosivuista 28,5 % käytti Flash Playeriä (29). Vuonna 2017 käyttö oli enää vajaa 6 % ja Silverlightin 0,1 % (21; 22). Microsoftin nykyisen, vuonna 2015 julkaistun Windows 10 -käyttöjärjestelmän oletusselain Microsoft Edge ei enää tue Silverlight-tekniikkaa (23; 24).

## **4 HTML5:n uudet elementit: <canvas> ja <video>**

### 4.1 HTML5 <canvas> -elementti

HTML5-tekniikan uusiin mediaelementteihin kuuluvalla monikäyttöisellä canvas-elementillä voidaan toteuttaa asioita, joissa vain käyttäjän taidot ovat rajana. Aiemmin tämä kaikki ei ollut mahdollista, vaan grafiikan piirtoon tai animaatioiden luontiin suoraan

selaimessa tarvittiin kolmannen osapuolen liitännäinen, joka yleensä oli Adobe'n Flash Player tai Microsoftin Silverlight. (6.)

HTML5:n <canvas>-elementtiä käytetään verkkosivuilla piirtämään grafiikkaa sivulle samanaikaisesti, niin sanotusti lennossa, ja yleensä elementtiä käytetään JavaScript-ohjelmointikielen avulla. Elementtinä canvas on vain grafiikan "säiliö", eikä sillä ole itsessään piirto-ominaisuutta, vaan piirtämiseen tarvitaan ohjelmointiskriptiä.

Canvas-elementtiä voidaan käyttää polkujen, graafisten elementtien ja tekstin piirtämiseen, tai siihen voidaan lisätä kuva-, video- ja audiotiedostoja useilla tavoilla. Lisäksi canvas-elementissä on monipuoliset animaatiomahdollisuudet, jolloin jokainen sivulla oleva objekti voidaan animoida. Käyttäjä tai kehittäjä voi luoda melkein mitä tahansa käytettäväksi canvas-elementissä, mikä tekee sen erittäin joustavaksi ja monipuolisesti käytettäväksi elementiksi. Canvas antaa oletusarvona jokaiselle piirretylle elementille väriksi mustan. Muita värejä elementeille voidaan antaa RGBA- (red, green, blue ja alfa-kanava) ja heksa-arvoina (heksadesimaalijärjestelmä). (66.)

Yksi HTML5:n canvas-elementin tärkeimmistä piirteistä on se, että se on täysin interaktiivinen: se pystyy tunnistamaan käyttäjän näppäimistökomentoja, hiiren klikkauksia tai kosketusta näytöllä, ja sen avulla voidaan ohjelmoida elementti reagoimaan näihin halutulla tavalla (26).

Toinen merkittävä ominaispiirre on kuvien tuottaminen canvas-elementtiin dynaamisesti, mikä voidaan toteuttaa suoraan palvelimelta tietokannasta, selaimen ohjelman tai käyttäjän syötteen avulla. Tämä tekee mahdolliseksi sen, että kuva voi olla muuttuva. (20, s. 201.)

Canvas-elementti määritellään HTML-koodiin leveys- ja korkeusattribuuteilla sekä id-määritteellä, joka antaa elementille tunnisteen JavaScriptiä varten. Sivulla voi olla useita canvas-elementtejä, ja ne ovat toisistaan riippumattomia. Elementteihin viitataan get(Context)-metodilla. Piirtäminen tehdään JavaScript-koodilla käyttäen Canvas API -sovellusliittymää, jolla luodaan konteksti (context): olio, jonka metodeilla piirretään (20, s. 202–203). Canvas API eli HTML5 Canvas 2D Context, on nimensä mukaisesti

kaksiulotteinen piirtopohja, jota käytetään JavaScriptillä (20, s. 203). 3D-grafiikkaa taas piirretään yleensä WebGL:n (Web Graphics Library) avulla (51).

Tällä hetkellä HTML5 canvas on hyvin tuettu kaikissa yleisissä selaimissa ja useimmissa mobiililaitteissa (kuva 4) (52).



Kuva 4. HTML5-tuki canvas-elementille eri selaimissa (52).

Canvas-elementin käyttö on yleistynyt tasaisen nopeasti, koska HTML5 on ollut web-suositus jo vuodesta 2014 lähtien. Toisin kuin aiemmin käytössä olleiden Adobe Flash Playerin ja Microsoftin Silverlightin, sen teknologia on avointa. Perustyökaluiksi tarvitaan vain selain ja koodieditori. Kehitystyötä helpottavat myös useat ilmaiset kirjastot. Lisäksi kerran kehitettyä toimintoa voidaan käyttää kaikissa alustoissa tietokoneista mobiililaitteisiin, joten se on hyvin monikäyttöinen.

Canvas-elementti on hyvin monipuolinen, sitä käytetään esimerkiksi 2D- ja 3D-pelien tuottamiseen, mainostamisessa korvaamaan Flash-pohjaisia bannereita ja ilmoituksia, erilaisten tietojen esittämiseen keräämällä dataa globaaleista lähteistä ja esittämällä sitä interaktiivisissa graafeissa. Koulutus- ja opetusikäytössä HTML5 canvasta voidaan käyttää yhdistämään tekstit, kuvat, videot ja audio erilaisiksi tehokkaiksi oppimiskokemuksiksi. (6).

## 4.2 HTML5 <video> -elementti

Yksi HTML5-tekniikan mediaelementeistä on <video>-elementti, jolla voidaan määritellä standardinomainen tapa upottaa videoita verkkosivuille. Videoelementti korvaa osittain aiemmin käytössä olleen object-elementin. Videoelementti kuuluu HTML5:n mediaelementteihin audio-elementin kanssa. Videoelementistä on tarkoitus tehdä uusi standardi, joka korvaa internetissä videoiden katselussa tarvittavan Adobe Flash Player -liitännäisen. (40.)

### Selaintuki videoformaateille

HTML5-tekniikka tukee kolmea videoformaattia: MP4:ää, WebM:ää ja Oggia (30). Kuvassa 5 esitetään, millainen tuki formaateille on eri selaimissa.

| Selain            | MP4 | WebM | Ogg |
|-------------------|-----|------|-----|
| Internet Explorer | x   | -    | -   |
| Chrome            | x   | x    | x   |
| Firefox           |     | x    | x   |
| Safari            | x   | -    | -   |
| Opera             | x   | x    | x   |

Kuva 5. Selaintuki HTML5:n eri videoformaateille (30).

Selainten erilaisen formaattien tuen takia kannattaa HTML-koodiin upottaa video vaihtoehtoisina tiedostomuotoina, jotta se näkyy varmasti eri selaimilla, ja lisäksi teksti, jos selain ei tue video-tagia ollenkaan (kuva 6). Selain toistaa videoversioista ensimmäisen, jonka se tunnistaa. Videotiedostossa on hyvä olla käyttäjää helpottavat ohjaimet (controls). Ohjaimilla käyttäjä voi säätää videon käynnistystä, pysäytystä tai äänen voimakkuutta. Myös kokoasetukset on hyvä asettaa häiriöiden välttämiseksi, vaikka selain useimmiten tunnistaa videon mitat automaattisesti.

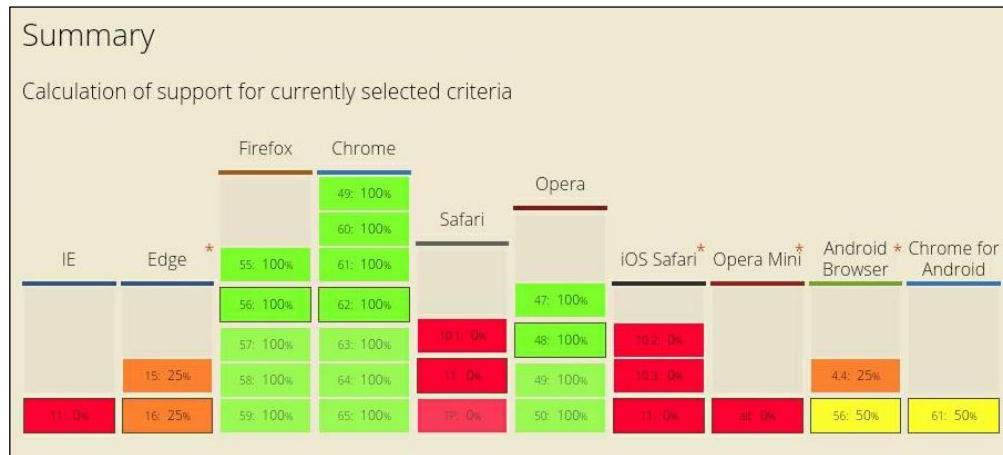
```
1 <video width="320" height="240" controls>
2   <source src="videofile.mp4" type="video/mp4">
3   <source src="videofile.ogg" type="video/ogg">
4   <source src="videofile.webm" type="video/webm">
5   Your browser does not support the video tag.
6 </video>
7
8 |
```

Kuva 6. Eri videoformaatit HTML-koodin videoelementissä.

Selaimen latautuvalle videolle voi olla hyvä asettaa muitakin erityismäärittelyjä sen mukaan, miten halutaan sen käynnistyvän. Autoplay-asetuksella video käynnistyy automaattisesti, ja loop-asetus toistaa videota yhä uudestaan. Ennen näiden asetusten käyttöä kannattaa harkita, tuoko se haluttua lisäarvoa videotiedoston näyttämiseksi. Huomioitavaa on myös, ettei Autoplay-asetus ei ole tuettu kaikissa mobiiliselaimissa (20, s. 182–189; 30).

## WebM

WebM on Mozillan, Operan, Adoben ja Googlen kehittämä videotiedostomuoto, joka on HTML5:n tukema (31). WebM on avoin, kaikkien käytössä oleva tiedostomuoto, jonka käytöstä ei tarvitse maksaa lisenssimaksuja. WebM perustuu Matroska-säiliöön, ja se on pakattu VP8- tai VP9-videokoodekilla ja sen sisältämä ääni Vorbis- tai Opus-koodekilla. (34; 35) WebM on optimoitu erityisesti videoiden korkealaatuisen reaaliaikaiseen internetkäyttöön. (35). Kuvassa 7 esitetään miten WebM on tuettu eri selaimissa.

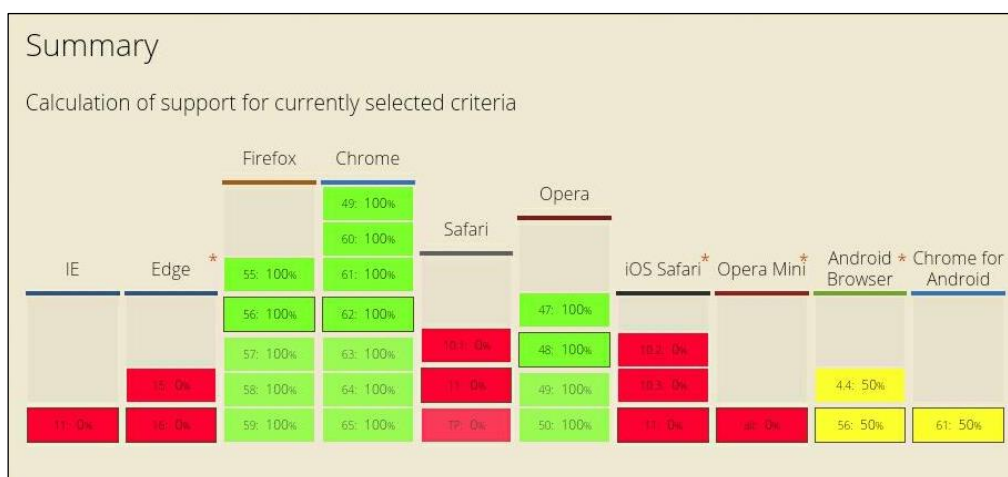


Kuva 7. HTML5-tuki WebM:lle (53).

## Ogg

Theora Ogg on Xiph.Org-säätiön kehittämä avoimen koodin multimediasäiliötiedostomuoto. Ogg on yksi HTML5-tekniikan tuetuista videoformaateista.

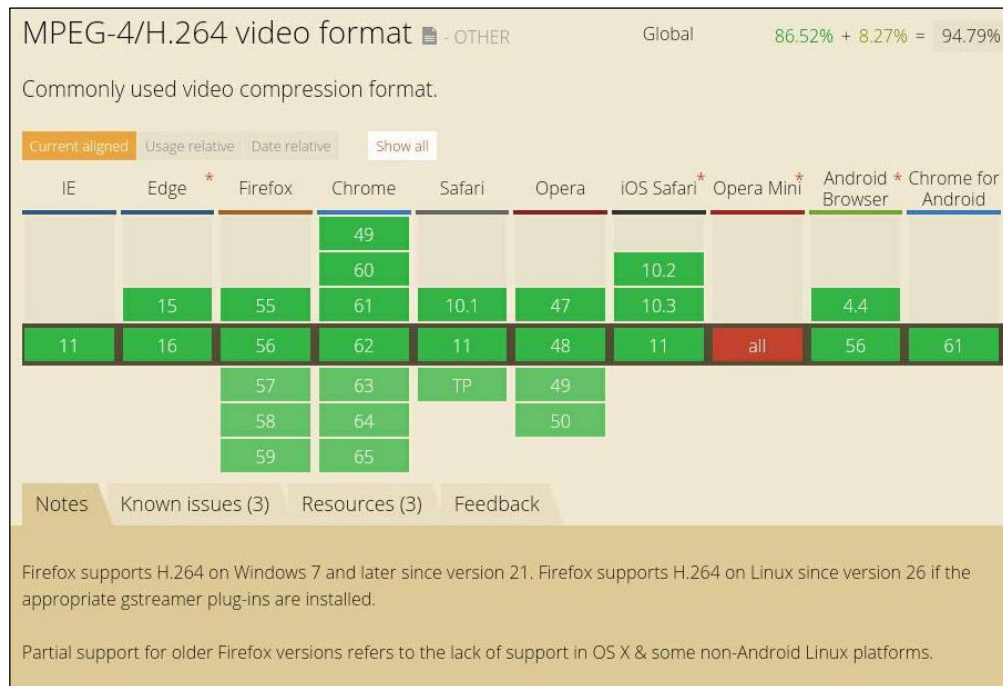
Ogg on striimaukseen perustuva säiliö, eli se voi kirjoittaa ja lukea tietoa samanaikaisesti. Tämä tekniikka tekee siitä erittäin sopivan internetkäyttöön (kuva 8), toisin kuin muut tiedostopohjaiset säiliömuodot. (31; 32.)



Kuva 8. HTML5-tuki Ogg-tiedostomuodolle (54).

## h.264

H.264 eli MPEG-4 AVC (Advanced Video Coding) on Moving Pictures Expert Groupin kehittämä videopakkausstandardi, joka perustuu Applen QuickTime-tiedostomuotoon. Tiedostomuoto on hyvin yleisesti käytetty videokameroissa ja tv-laitteissa. H.264 on tuettu kaikissa HTML5-selaimissa, kuten kuvassa 9 esitetään. (31; 33.)



Kuva 9. HTML5-tuki H.264:lle (55).

AVC-videokoodekki (codec) käyttää uuden sukupolven häviöllistä pakkaustapaa. Videokoodekki on kehitetty yhteistyössä ITU-T Video Coding Experts Group (VCEG)- ja ISO/IEC Moving Picture Experts Group (MPEG) -työryhmien kanssa. AVC-videokoodekki lisää uusia tekniikoita verrattuna vanhempaan MPEG-4 part 2 -pakkaustekniikkaan, mutta tekniikat ole eivät kuitenkaan ole yhteensopivia. (33.)

### Decoding

Videokoodekeilla pakataan ja puretaan videotiedostoja. Koodekkeja tarvitaan, koska videotiedostot ovat suurikokoisia ja ilman pakkausta niiden liikuttelu internetissä on hidasta ja hankalaa. Pakkaustapa on useimmiten häviöllistä, ja siinä yhdistetään sekä

kuvan että liikkeen pakkausta. Vaikka pakkaus on häviöllistä, pyritään videosta pakkaamalla poistamaan sellaista informaatiota, jota ihmissilmä ei erota. (44.)

## 5 HTML5 video + canvas ja pikselimanipulaatio

Yhdistämällä HTML5-tekniikan canvas- ja videoelementit saadaan magiikkaa, näin väittää HTML5-tohtori Tab Atkins Jr. Canvas- ja videoelementit on suunniteltu toimimaan yhdessä, ja testaan Tab Atkinsin ohjeilla, mitä niillä voidaan tehdä. Toimiiko videon pikselimanipulaatio, kuten HTML5 Doctor -sivustolla annetuissa ohjeissa kerrotaan (46). Käytän ilmaista testaukseen tarkoitettua demovideotiedostoa ja tallennan siitä kolme HTML5-tekniikan tukemaa versiota: WebM-, Ogg- ja h.264-tiedostot. Lisään videotiedostoihin ohjaimet (controls), joilla videota voidaan esittää ja pysäyttää, kontrolloida äänen tasoa ja niin edelleen. Loop-toiminto toistaa videota yhä uudestaan lopusta alkuun (kuva 10).

```

1  <!DOCTYPE html>
2  <title>Video/Canvas Demo 1</title>
3
4  <canvas></canvas>
5  <video controls loop>
6      <source src=video.webm type=video/webm>
7      <source src=video.ogv type=video/ogg>
8      <source src=video.mp4 type=video/mp4>
9  </video>
10 </html>

```

Kuva 10. Esimerkki videotiedostoista canvas-elementissä.

Selaimen pitää tukea HTML5:n <video>-elementtiä, ja testivideot toimivat hyvin Firefoxissa ja Chromessa, mutta Internet Explorer ei sitä oikein tue versiossa 8 ja sitä aiemmissa.

Testaan testissä kolmea asiaa:

1. Ensimmäiseksi testaan, että video toimii canvas-elementin kanssa.
2. Toisessa testissä video toistuu edelleen keskellä, mutta canvas täyttää koko selaimen ikkunan ja siinä toistuu JavaScriptillä tehty suurennos videosta. Selain osaa skaalata toisen version jokaisesta videon kuvaruudusta selaimen ikkunan kokoiseksi "taustaksi". Tämä näkyy kuvassa 11, joka on ruutukaappaus videosta.



Kuva 11. JavaScriptillä tehty videon suurennos täyttää koko selainikkunan canvas-alueen.

3. Kolmanneksi testaan suoraan selaimessa videon pikseleiden käsittelyä eli sitä, miten voin muuttaa videon mustavalkoiseksi canvas-alueella. Tässä pikseleiden käsittely tehdään kaappaamalla ensin kuvan sisältö, muuttamalla sitä ja sitten piirtämällä se takaisin canvas-alueelle.

Kun video laitetaan päälle, selain tekee siitä toisen version mustavalkoisena viereen. Tämä on esitetty kuvassa 12.

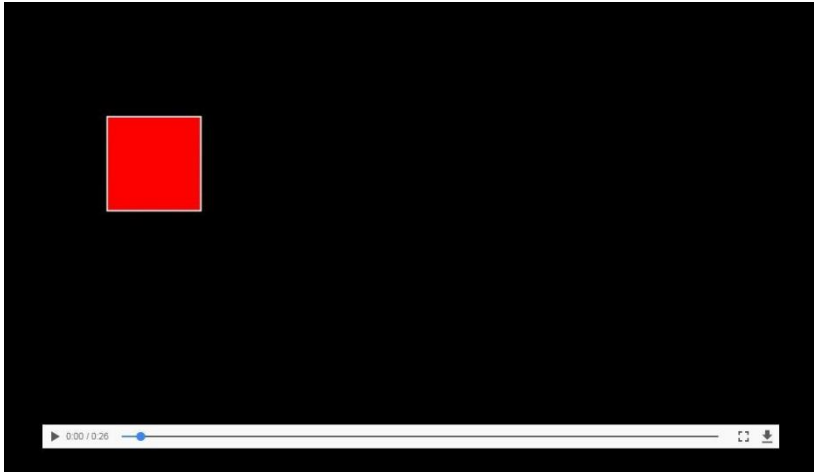


Kuva 12. Alkuperäinen video vasemmalla ja oikealla pikselimanipulaatio canvas-alueella.

Canvas-elementti itsessään on tyhjä, ja siihen pitää laittaa sisältöä tai muotoiluja CSS-tyylitiedostolla. Canvas-elementti on myös resoluutioriippuvainen, joten renderöinnillä tehty skaalaus ei aina mene täysin oikein. Pikselimanipuloidusta videon mustavalkoisesta versiosta kuvassa 12 huomataan, etteivät mittasuhteet toistu samoin kuin vieressä olevassa alkuperäisessä videossa. (66.)

Tähän mennessä videon pikselimanipulaatio näyttää toimivan yllättävän helposti, eikä tarvitse osata juurikaan itse koodata, koska valmiita koodeja löytyy kokeiltavaksi ja niitä voi muokata omiin tarpeisiin.

Seuraavaksi haluan testata, miten saan canvas-alueelle piirretyn liikkuvan elementin taustan läpinäkyväksi, jolloin taustalla oleva kuva tulee näkyviin. Liikkuvana elementtinä käytän Codeproject.com-sivulta ladattua valmiiksi tehtyä testivideota, jossa liikkuu punainen neliö (kuva 13).



Kuva 13. Codeproject.com-sivuston video punaisesta liikkuvasta neliöstä (67).

Testissä lisään kuvan videon taakse ja muutan videon mustan taustan läpinäkyväksi, jolloin punainen neliö liikkuu paikallaan pysyvän kuvan päällä, kuten kuvassa 14 esitetään.



Kuva 14. Videon punainen neliö liikkuu ja taustalla oleva kuva näkyy läpi (67).

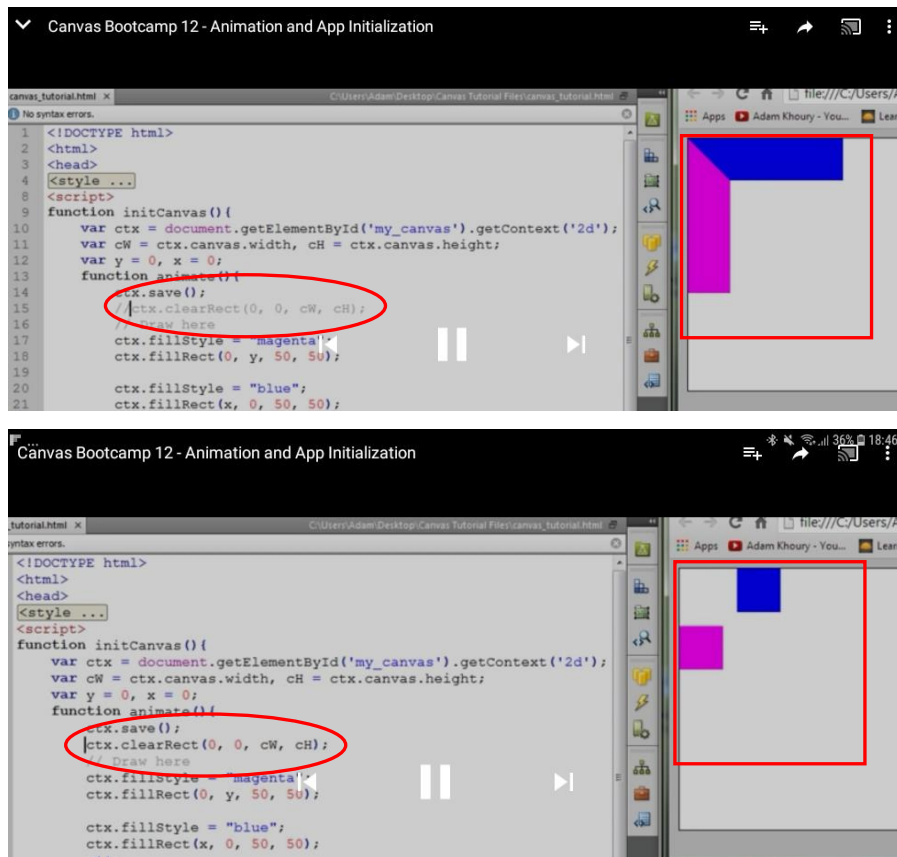
HTML5:ssä tiedon tallentamiseen ja käyttämiseen käytetään web-muistia, joka on paljon kehittyneempi kuin käytössä olevat web-istunnon evästeet (cookies). Web-muisti tallentaa käyttäjän selaimen erilaisia tietoja, joita se käyttää uudelleen näyttämään käyttäjälle henkilökohtaisia näkymiä, esimerkiksi millä sivuilla käyttäjä on käynyt tai mitä toimintoja käyttäjä on valinnut vaikkapa tehdessään ostoksia verkkokaupoissa. (20, s. 228–230). Web-muistia voidaan käyttää myös offline-tilassa ja sen kautta lukea esimerkiksi palvelimelta ladattuja tiedostoja (59).

HTML5:ssä on myös tuki geolokaatioon (Geolocation API), eli käyttäjän sijainti pystytään selvittämään, jos käyttäjä on antanut siihen luvan. Tekniikkaa voidaan käyttää personoitujen mainosten ja sisällön näyttämiseen tai vaikkapa paikkakuntaakohtaisen säätiedon esittämiseen. (20, s. 237–238.)

## 6 Testityö ja työn vaiheet

Insinööriyönä tehdyn testityön tarkoitus oli selvittää, kuinka hyvin voidaan luoda HTML-sivu, joka sisältää personoidun videon, ilman sen syvempää koodausosaamista ja käyttämällä yleisiä standardeja ja kirjastoja. Personoidulla videolla tarkoitetaan tässä videota, jonka päälle on liitetty liikkuvaa kuvaa, esimerkiksi tuotekuvaa, niin että vaikuttaa siltä, kuin video olisi yksi kokonaisuus. Työkaluina käytettiin pelkkiä yleisiä standardikieliä (HTML5, JavaScript ja CSS) ja tarvittaessa eri kirjastoja ja metodeja.

Peruseriaate pikselimanipuloidun videon sisältävän HTML-sivun rakentamisessa on se, että kun sivulle ladataan haluttu video, siitä otetaan jokaisesta kuvaruudusta (frame) ”ruutukaappaus” joka tallennetaan olioon, joka sisältää piirto-ominaisuudet. Tämän jälkeen voidaan videota käsitellä halutusti, esimerkiksi piirtämällä toista kuvaa videon päälle, ja lopputulos piirretään edelleen merkitylle canvas-alueelle. Canvas-alue tyhjenetään jokaisen ruudun piirtämisen jälkeen, ja koko prosessi toistetaan nopeasti peräkkäin niin, että lopputuloksena on animaation kaltainen vaikutelma. Mikäli canvas-aluetta ei tyhjenetä jokaisen piirtämisen jälkeen, jää piirretty ruutu näkyviin ja seuraavassa kohdassa näkyy uusi kopio, jolloin liike-efektiä ei synny. Kuvassa 15 näytetään tämän ero sekä koodirivillä että ruudulla.



Kuva 15. Vaikutus canvas-alueen tyhjentämättä jättämisellä ja tyhjennyksellä (68).

## 6.1 HTML-sivun perusrakenne

Testityön koodi jaettiin kolmeen osioon: itse HTML-koodiin, CSS-tyylitiedostoon ja JavaScript-ohjelmointikoodiin. Tämä jako näkyy esimerkkikoodissa 1. Koska koodi ei ole kovin pitkä, rakenteen selkeyden vuoksi kaikki koodiosiot pidettiin samalla HTML-sivulla, vaikka yleensä on järkevää pitää koodit omissa erillisissä tiedostoissaan: `.css`-tiedosto tyylejä varten ja `.js`-tiedosto JavaScript-koodia varten.

Itse testityön HTML-koodin perusrakenne oli hyvin yksinkertainen ja tyylimuotoilu jätettiin vähäiseksi, jotta koodin peruslogiikka tulisi mahdollisimman selkeästi esiin.

```
<!DOCTYPE HTML>
<html>
<head>
<title>Canvas: HTML5</title>
<meta charset="utf-8">
<link href="myStyle.css" rel="stylesheet" />
<script src="myScript.js"></script>
</head>
<body>
  <canvas id="myCanvas" width="640" height="360">
    Upgrade Browser
  </canvas>
  
  <video id="video" src="xxx.mp4" style="display:
  none;"></video>
</body>
</html>
```

Esimerkkikoodi 1. Yleinen HTML-sivun rakenne.

## 6.2 Videon kuvaaminen ja HTML-tiedostoon lisääminen

Testityötä varten kuvasin useita videoita selvittääkseni, mikä tapa toimisi lisättävän kuvan kanssa parhaiten. Kuvaukseen käytin Samsungin älypuhelinta, koska asetuksissa saadaan valittua tarpeeksi laadukasta teräväpiirtokuvaa. Asetuksina käytin

- kuvakokoa 4:3
- resoluutiota Full HD (60 fps) 1920 x 1080 pikseliä.

Jotta saisin lisättävän kuvan parhaiten paikoilleen kuvassa liikkuvaan objektiin, objektin pitäisi liikkua mahdollisimmat tasaisesti. Koska puhelimeen ei saa jalustaa kiinnitettyä, hylkäsin kuvaustavan, jossa liikkuisin itse eteenpäin, liian tärähtelevänä. Kuvasin useita liikkuvia busseja ja pakettiautoja. Osan videoista jouduin hylkäämään eteen tulleiden autojen tai ihmisten takia, koska en pystyisi poistamaan niitä videosta ja lisätty kuva tulisi niiden päälle. Parhaassa videossa näkyikin tällainen kauneusvirhe tienviitan muodossa toisessa reunassa ja lisätty kuva peittää sen. Useissa kuvaamissani videoissa oli myös epätasaista liikkumista, auto joko kiihdytti tai jarrutti. Tämä liike hankaloittaa, koska se pitäisi ottaa huomioon skriptissä, jotta lopputuloksen liike olisi mahdollisimman luonnollinen.

Videon lisääminen HTML-tiedostoon tehtiin <video>-tagillä:

```
<video id="video" src="truck.mp4"
  style="display: none;"></video>
```

Tällä koodiosalla annettiin videolle id sekä määrittely, ettei videota näytetä vielä sivulla. Video piirretään myöhemmin <canvas>-piirtoalueelle.

### 6.3 Sivun lataaminen ennen toimintoja ja piirtoalueen lisääminen

Ensimmäiseksi skriptiosassa ennen muita toimintoja annetaan sivun ensin ladata tarvittavat tiedot. Tämä tehdään määritteellä

```
window.onload = function() {};
```

Näin varmistetaan, ettei piirtämistä aloiteta, ennen kuin koko videotiedoston sisältö on latautunut sivulle, jottei videon näyttämisessä tule viiveitä tai katkoksia.

Videon muokkausta varten määritellään sivulle <canvas>-elementti. Se on alue, johon videon kuvaruutuja piirretään.

### 6.4 Olion tekeminen

Ennen kuin canvas-alueelle voidaan piirtää mitään, tarvitaan olio, joka sisältää sekä kontekstin että tarvittavat piirto-ominaisuudet. Tämä on toteutettu seuraavasti:

```
document.getElementById("demo");
```

Ensin haetaan muuttujaelementti:

```
var myCanvas document.getElementById("myCanvas");
```

Se viittaa canvas-elementtiin, jonka id on myCanvas.

Luodaan kontekstimuuttuja, joka viittaa myCanvas-muuttujaan ja otetaan sen konteksti mukaan.

```
var ctx = canvas.getContext(contextType);
```

Testityössä getContext-metodi käytti arvoa 2d, joka tarkoittaa kaksiulotteista sisältöä.

Seuraavaksi luodaan kontekstille olio:

```
var context = myCanvas.getContext('2d')
```

Tällä tavalla saadaan olio, joka sisältää kaikki piirtämisen ominaisuudet ja funktiot, joita tarvitaan siihen, että canvas-alueelle voidaan piirtää.

## 6.5 Olion piirtäminen ja lataaminen

Konteksti-olio sisältää metodin nimeltään Drawimage(). Metodien avulla voidaan piirtää kuva tai video canvas-alueelle:

```
context.drawImage(myVideo, x, y, w, h);
```

Testityössä piirrettiin kahta elementtiä: videon ruutukuva taustalle ja sen jälkeen kuva videotaustakuvan päälle.

Ennen kuin oliota voidaan käyttää, sen pitää olla ladattu. Tämä saadaan aikaan onload.-komennolla:

```
object.onload = function(){myScript}; (JavaScript).
```

## 6.6 Aikavälit ja piirtoalueen tyhjennys

Koska canvas-alueelle voidaan piirtää vain staattisia kuvia, toistetaan piirtämistä ja tyhjennetään piirtoalue niin, että aina valitun ajan välein otetaan uusi kuvaruutu videosta ja kuvasta ja määritellään sille mahdollisesti uudet koordinaatit liikkeen mukaan. Testityössä x-koordinaattia siirrettiin joka kerta 17 pikseliä 60 millisekunnin välein. Kun näin tehdään tarpeeksi usein ja nopeasti, luodaan illuusio liikkuvasta kuvasta.

Metodi setInterval() kutsuu funktiota määritellyllä aikavälillä, kunnes joko ikkuna suljetaan tai kutsutaan metodi clearInterval():

```
setInterval(function, milliseconds, parameter1, parameter2,
...)
```

Kun piirto on suoritettu canvas-alueelle ja halutaan piirtää se uudestaan, kannattaa ensin tyhjentää piirtoalue. Jos sitä ei tehdä, piirrosta syntyy uusi kopio edellisen viereen, kuten tämän luvun alussa kuvassa 15 on esitetty, eikä liike-efektin vaikutusta siirtyvästä kuvasta synny. Piirtoalueen tyhjennys tehdään metodilla `ctx.clearRect()`:

```
context.clearRect(x, y, width, height);
```

## 6.7 Liikkuva tuotokuva videossa

Siinä, missä video pysyy sivulla samassa paikassa, tuotokuva, joka sijoitetaan videon päälle, pitää saada liikkumaan videokuvassa liikkuvan kohde-elementin mukana. Se on saatu aikaiseksi muuttamalla päälle sijoitettavan kuvaelementin koordinaatteja ja kokoa (aloituskoordinaatit sekä leveys- ja korkeusmitat). Tämä joudutaan mahdollisesti tekemään joka kerta, kun kuva piirretään uudestaan, mikäli kuvan sijainti tai koko muuttuvat aiemmasta.

Koska testityössä pyrittiin testaamaan tekniikkaa ja mahdollisimman tyylikkään ja huomaamattomasti videoon istuvan kuvan koodaaminen käsin osoittautui HTML-koodissa hieman haastavaksi, oli järkevintä pyrkiä kuvaamaan video niin, että päälle piirrettävää kuvaa liikutettaisiin mahdollisimman tasaisesti vaakasuunnassa ja se pysyisi samassa kohdassa pystysuunnassa. Testivideo oli myös hieman huonosti rajattu, sillä vaihtuva kuva peittää reunassa auton edessä olevan liikennemerkkin.

Ensin testattiin JPG-muotoista kuvaa, joka tallennettiin kuvapankkiin ja linkki sijoitettiin HTML-koodiin (kuva 16).



Kuva 16. JPG-muotoinen kukkakuva liikkuvan pakettiauton kyljessä.

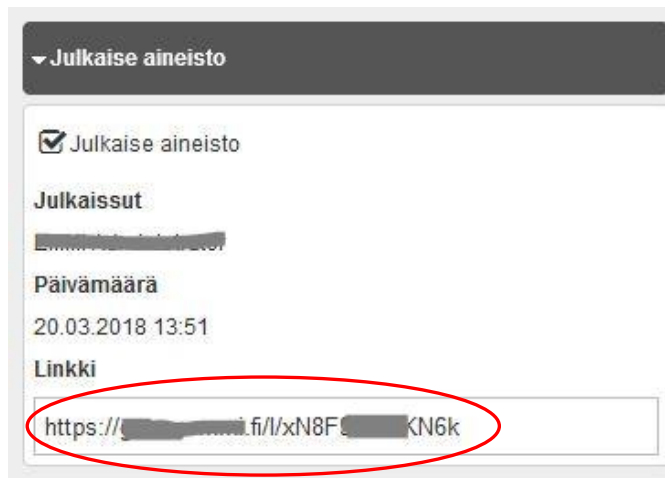
Seuraavaksi vaihdoin kuvan kuvapankissa PNG-muotoiseen kuvaan, joka tukee läpinäkyvyyttä. Käynnistin videon uudelleen, ja vaihdettu kuva vaihtui myös videoon ilman koodiin koskemista (kuva 17).



Kuva 17. PNG-muotoinen kukkakuva liikkuvan pakettiauton kyljessä.

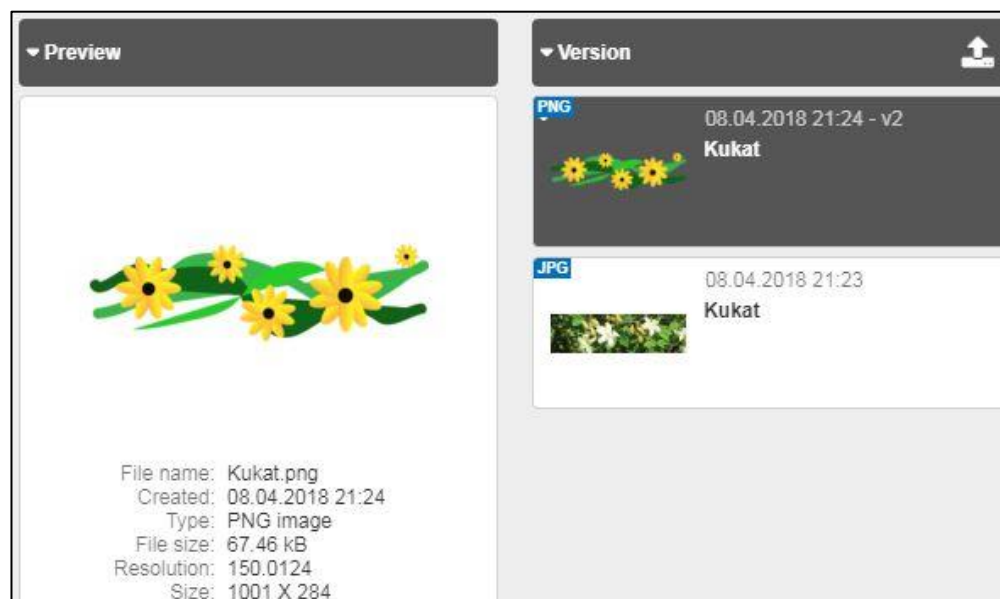
Koodissa käytetään src-attribuuttia, jolla määritellään paikka, jossa haluttu video- tai kuvatiedosto sijaitsee. Src-attribuutista voisi tehdä myös muuttujan, mutta on yksinkertaisempaa käyttää URL-sijaintia ja huolehtia siitä, että attribuutilla määritelty tiedosto on ylläpidettävissä ja muutettavissa toisessa paikassa linkin takana. Aiemmin työssä esitetty kuvapankki (DAM) voi hyvin toimia tällaisena paikkana. Testityössä sain käyttää erään suuren sisältöpalvelujen tuottajan kuvapankkia testaukseen.

Kuvapankissa voi olla käytettävissä myös versiohallinta ja ominaisuus, jonka avulla luodaan latauslinkki (kuva 18) itse aineistoon.



Kuva 18. Tiedoston julkaisu kuvapankista.

Latauslinkki lisätään HTML-koodiin src-attribuutiksi, ja muuttamalla kuvapankissa kuvaa (kuva 19) itse tiedosto myös muuttuu käytettäessä tätä samaa linkkiä, tässä tapauksessa HTML-koodissa olevaa linkkiä. Tätä versiohallintaominaisuutta testasin vaihtamalla ensimmäisen JPG-muotoisen pikselikuvan PNG-muotoiseen, jonka tiedostomuoto tukee kuvassa olevaa läpinäkyvyyttä.



Kuva 19. Tiedoston versiohallinta. Vasemmalla esikatselussa näkyy tällä hetkellä käytetty kuva.

Kuvapankin lisäksi voidaan käyttää toista ratkaisua, jolloin tiedosto haetaan ”lennossa” pilvestä käyttäen hajautettua sisällönjakeluverkostoa (Content Distribution Network). CDN-ratkaisun hyöty on, että kuvat ja videot voidaan hakea pilvestä riippumatta siitä, toimiiko DAM-ratkaisu. CDN-ratkaisun seurauksena tiedoston haku on myös mahdollisimman nopeaa. Sisällönjakeluverkko, CDN, on maantieteellisesti hajautettu verkko, joka koostuu välityspalvelimista ja niille tarpeellisista konesaleista ja yhteyksistä. Tällaisen sisällönjakeluverkon tarkoitus on palvelujen tarjoaminen loppukäyttäjien lähellä, jotta saadaan aikaiseksi parempi toimintavarmuus pienemmällä viiveellä.

## **7 Tekniikan kehittyminen liikkuvan kuvan muokkaukseen**

### **7.1 HTML5 yleisstandardina ja sen tulevaisuus**

Vuonna 2010 ajateltiin, että HTML5:n standardointi web-tekniikaksi ei ole valmis vielä vuosiin tai jopa kokonaiseen vuosikymmeneen. Vuonna 2014 siitä tuli kuitenkin W3C:n suositus, ja elokuussa 2017 julkaistiin HTML 5.2 W3C:n suositusehdokkaaksi. HTML5 on käytössä noin 67 prosentissa verkkoselaimia, ja vanhempaa HTML-tekniikkaa käyttää noin 78 prosenttia verkkoselaimista. (25; 56 ja 57.)

HTML5:n suosio perustuu sen moniin uusiin ominaisuuksiin, kuten responsiivisen web-suunnittelun mahdollisuuteen: Kehittäjille on ollut haasteena mobiililaitteiden, kuten älypuhelimien ja tablettien, kasvava osuus verkkosivujen käytöstä. HTML5-tekniikan etu on mahdollisuus responsiiviseen internetsivustojen tekoon, jolloin tekniikka tekee mahdolliseksi verkkosivujen käytön millä laitteella tahansa, ja ne näyttävät aina hyvältä, mikä on tärkeä asia eri sovellusten valmistajille ja tietenkin markkinoijille. Responsiivisuudella tarkoitetaan sivustojen mukautuvuutta eri laitteisiin. Sivusto mukauttaa automaattisesti laitteen mukaan sisällön, ulkoasun ja sivulla käytetyt toiminnallisuudet, jolloin riittää yksi sivusto ja sisältö kaikkiin laitteisiin. (47; 58 ja 59).

Uusien mobiilisovellusten kehittäminen sekä Applen iOS- että Android-järjestelmiin on ollut haastavaa, sillä kehittäjät ovat joutuneet tekemään oman versionsa kummallekin käyttöjärjestelmälle. Pienemmillä yrityksillä ei ole tähän resursseja, ja ne ovatkin ottaneet mieluummin HTML5-tekniikan myötä tyytyväisenä natiivisovellukset käyttöön. Käyttämällä ja yhdistämällä HTML5-tekniikka, CSS3-tyylitiedosto ja JavaScript-ohjelmointikieli voidaan rakentaa yksi versio ohjelmista, ja se toimii tietokoneilla, mobiililaitteilla, eri selaimilla ja eri laitealustoilla. Näin alustariippumattomien sovellusten kehittäminen helpottuu HTML5-tekniikan ansiosta. (47.)

Lyhyessä ajassa isot web-tekniologiayritykset, kuten Google ja Facebook, ovat vaihtaneet videoformaattinsa tuen Flashistä HTML5-videoihin. Suomessa Yle otti HTML5-tuen käyttöön Arenassa syksyllä 2017. Trendi näyttää jatkuvan, koska HTML5-muotoiset videot ovat käyttäjäystävällisempiä nopeuden ja natiivimuotonsa takia. (47; 48 ja 58.)

HTML5-tekniikan yksinkertainen koodi tekee sivujen kehittäjille helpommaksi muuttaa ja ylläpitää ohjelmia. Tämä tuo samalla kustannussäästöjä ja kilpailuetua.

Myös sivustojen käyttäjäystävällisyys on tärkeä asia, ja käyttäjät odottavat sivujen latautuvan nopeasti ja olevan käytettävissä kaikilla laitteilla kaikkialla siellä, missä käyttäjäkin liikkuu. Ohjelmien kehittäjille ja mainostajille HTML5-tekniikan paikannuspalvelut muun muassa Geolocation API:n avulla tuovat uusia etuja, koska mainoksia voidaan personoida käyttäjän sijainnin perusteella. Kehittäjät taas haluavat luoda mahdollisimman houkuttelevia internetsivuja, mutta mitä enemmän visuaalisia elementtejä, kuten kuvia, grafiikkaa ja multimediatiedostoja, sivu sisältää, sitä raskaampi se on käyttää, ellei käytetä HTML5-tekniikkaa, jolloin ei tarvita kolmannen osapuolen liitännäisiä, vaan grafiikkaa voidaan luoda lennossa canvas-elementin avulla. (47.)

HTML5-tekniikan suosio ja halu sen käyttöön tulevaisuuden tekniikkana näkyy muun muassa siinä, että nyt keskitytään siihen, että HTML5-tekniikkaa kehitetään ja sitä päivitetään joka vuosi eikä joka 10. tai 15. vuosi. Erityisesti kehitetään uudenlaisia mahdollisuuksia videotiedostojen näyttämiseen, esimerkiksi Encrypted Media Standardia (EME), joka tarjoaa sovelluksen, jolla voidaan hallita oikeuksia, kuka videoita voi katsoa. (60.)

## 7.2 Edut ja mahdollisuudet

Kun liikkuvaa kuvaa voidaan muokata ja personoida, markkinoijat pystyvät paremmin tuottamaan personoitua markkinointia kaikissa käyttämässään kanavissa ja sitä kautta luomaan kokonaisvaltaisen personoidun kokemuksen kuluttajalle. Nykyään kuluttajat käyttävät erilaisia kanavia pirstoutuneesti ja sattumanvaraisesti, jolloin on tärkeää, että personoitu markkinointi näkyisi samalla tavalla kaikissa kuluttajan käyttämässä kanavissa.

Kun tavat ja tekniikat tietojen keräämiseen kuluttajasta koko ajan kehittyvät ja tietoa voidaan analysoida entistä tehokkaammin, pystytään tuottamaan enemmän olennaista markkinointia kuluttajalle. (62.)

### Integraatiot (REST/SOAP)

Erilaiset järjestelmät ja palvelut tarjoavat entistä enemmän mahdollisuuksia jakaa tietoa ja palveluita muiden järjestelmien kanssa. Tähän voidaan käyttää esimerkiksi REST/SOAP-rajapintaa, jolloin voidaan automatisoida ja jakaa esimerkiksi tuotetietoa PIM-järjestelmistä tai tuotekuvia DAM-järjestelmistä suoraan videotuotantoon. (13.)

## 8 Tulokset ja johtopäätös

Mainonnan personointi on kehittynyt nopeasti viimeisen 25 vuoden aikana. Tekniikka ja laitteet, kuten tietokoneet ja niiden ohjelmistot, painokoneet ja digitaaliset painokoneet sekä internetin ja HTML:n kehitys ovat tehneet mahdolliseksi tämän nopean kehittymisen. Tekniikan huiman kehityksen myötä on tullut mahdolliseksi tuottaa kustannustehokasta personoitua markkinointia, mikä vastaa sekä markkinoijien että kuluttajien tarpeisiin.

Viimeisten vuosien aikana mainosten personointi on liittynyt vahvasti videomainontaan ja videoiden kasvavaan suosioon. Videoiden toiminnallisuuksia kehitetään jatkuvasti, ja HTML5-standardi on tuonut uusia mahdollisuuksia videoiden muokkaamiseen. Tätä mahdollisuutta käytetään yhä enenevässä määrin mainonnassa ja markkinoinnissa,

koska on huomattu, että kuluttajat haluavat mieluummin itselleen personoitua mainontaa.

Insinööriyössä tutkittiin ja testattiin tätä trendiä sekä teoriassa että käytännössä. Testityötä tehdessä huomattiin, että liikkuvan kuvan käsittely oli osin yllättävän haasteellista, vaikka videon toisto, kuvan lisääminen canvas-alueelle ja muut perusasiat olivat helppoja oppia ja toteuttaa. Suurimpia ongelmia testityössä kuitenkin oli se, että canvas ei pysty käsittelemään videota ja siihen liittyvää aikaa kovinkaan helposti, joten piti itse laskea aika haluttuun kuvaruutuun. Tämä toi oman haasteensa erityisesti silloin, jos kuva ei liiku tasaisesti suoraan ylös tai alas tai kohde ei liiku tasaista vauhtia.

Mielenkiintoinen testityön osuus oli muuttuvan kuvan lisäys. Sitä testattiin kolmella tavalla: ensin kuva oli tiedostona omalla koneella koodin kanssa, toisessa kuva haettiin URL-linkin takaa kuvapankista ja kolmantena kuvaan viitattiin CDN:stä (pilvestä), jolloin kuvatiedoston hallinta oli hyvin helppoa koskematta koodiin. Jos kuva vaihdettiin pankissa vaikkapa aivan toiseen kuvaan ja toiseen tiedostotyyppiin, se päivittyi heti videoon ilman koodiin koskemista.

Vaikka HTML5:n käyttö videon personointiin oli suhteellisen helppoa ja vaivatonta, tultiin työssä johtopäätökseen, että jos muuttuvan kuvan ja tieto halutaan yhdistää visuaalisesti saumattomasti, tällä hetkellä paras olisi käyttää valmiita olemassa olevia ratkaisuja, jossa koodi ja kuvatiedoston liikkumisen haasteet on ratkaistu jo valmiiksi koodissa ja personoitu tieto istuu videoon saumattomasti. Tällöin myös lopputulos on käyttäjälle huomaamaton ja ammattimaisesti toteutettu.

## Lähteet

- 1 EasyCatalog for Adobe® InDesign®. 2017. Verkkodokumentti. <<https://www.65bit.com/software/easycatalog/>>. Luettu 26.8.2017.
- 2 Kirkpatrick, David. 2016. Study: 71% of consumers prefer personalized ads. Verkkodokumentti. <<http://www.marketingdive.com/news/study-71-of-consumers-prefer-personalized-ads/418831/>>. 9.5.2016. 26.8.2017.
- 3 Kohderyhmät ja osoitteet. 2017. Verkkodokumentti. Kopio Niini Oy. <<http://www.kopioniini.fi/suoramarkkinointi/kohderyhmat-ja-osoitteet/>>. Luettu 27.8.2017.
- 4 Suoramainonta. Verkkodokumentti. Suomen Mediaopas. <<http://www.mediaopas.com/suoramainonta/>>. Luettu 27.8.2017.
- 5 Voutilainen, Antti. 2015. Järjestelmät eivät ratkaise markkinoinnin ongelmiasi, ihmiset ratkaisevat. Verkkodokumentti. <<http://www.marmai.fi/blogit/vierasblogi/jarjestelmat-eivat-ratkaise-markkinoinnin-ongelmiasi-ihmiset-ratkaisevat-6295802>>. Päivitetty 4.2.2016. Luettu 12.9.2017.
- 6 Gerchev, Ivaylo. 2014. HTML5 Canvas Tutorial: An Introduction. Verkkodokumentti. <<https://www.sitepoint.com/html5-canvas-tutorial-introduction/>>. 10.3.2014. Luettu 18.8.2017.
- 7 Lazaris, Louis. 2010. HTML5 Canvas Element Guide. Verkkodokumentti. <<https://www.webpagefx.com/blog/web-design/canvas-element/>>. 12.10.2010. Luettu 19.8.2017.
- 8 Yleistä tietoa painotuotteista, painamisesta ja kuvanmuodostuksesta. Verkkodokumentti. PainoIndexi. <<http://painoindexi.fi/yleista-tietoa>>. Luettu 7.9.2017.
- 9 Johnson, Harald. 2004. Mastering Digital Printing. Edition2. Course Technology.
- 10 Painomenetelmät. 2015. Verkkodokumentti. Graafinen. <[www.graafinen.com/tietopankki/painomenetelmat](http://www.graafinen.com/tietopankki/painomenetelmat)>. 25.1.2015. Luettu 7.9.2017.
- 11 Kokey, Nathaniel. 2011. Technology's impact on Designers. Variable Data Printing. Verkkodokumentti. <<https://nathanielkok.wordpress.com/2011/11/10/5/>>. 10.11.2011. Luettu 28.9.2017.
- 12 What is Web to Print. 2017. Verkkodokumentti. OPS. <<http://www.onlineprintsolution.co.uk/what-is-web-to-print/>>. 28.4.2017. Luettu 2.9.2017.
- 13 Sirve, Olle. 2017. Senior Consultant, Grano Oy, Helsinki. Keskustelu 26.8., 3.9. ja 29.9.2017.

- 14 Cox, Barney. 2008. Adobe launches PDF Print Engine 2 to bring VDP printing into the mainstream. Verkkodokumentti. <<https://www.printweek.com/print-week/news/1123944/adobe-launches-pdf-print-engine-bring-vdp-printing-mainstream>>. 28.5.2008. Luettu 26.8.2017.
- 15 Viluksela, Pentti; Ristimäki, Seija & Spännäri, Toni. 2010. Painoviestinnän tekniikka. Helsinki: Opetushallitus.
- 16 Whitepaper. Verkkodokumentti. PDF/VT – the ISO Standard for Variable Data Printing (VDP) Applications. <<https://www.pdf-tools.com/public/downloads/know-how/whitepaper-pdfvt-standard-iso-16612-2-en.pdf>>. Luettu 27.8.2017.
- 17 Viimeinkin piristystä mediamainonnan investoinneissa – jatkuuko nouseva trendi 2017? Kantar TNS. Verkkodokumentti. <<https://www.tns-gallup.fi/uutiset/mainonnan-maara-2016>>. Luettu 13.9.2017.
- 18 Xerox launches internet printing solution. 1995. Telecompaper. Network World.
- 19 Jorij, Abraham. 2014. Product Information Management Theory and Practice. Springer International Publishing.
- 20 Korpela, Jukka K. 2012. HTML5 uudet ominaisuudet. Jyväskylä: Bookwell.
- 21 Global Media Format Report 2016. 2016. Verkkodokumentti. Encoding.com. <<https://www.encoding.com/files/2016-Global-Media-Formats-Report.pdf>>. Luettu 1.9.2017.
- 22 Usage of Flash for websites. 2017. Verkkodokumentti. W3Techs. <<https://w3techs.com/technologies/details/cp-flash/all/all>>. 30.10.2017. Luettu 12.9.2017.
- 23 Flacy, Mike. 2015. Microsoft won't include support for Silverlight in Windows 10 Edge browser. Verkkodokumentti. <<https://www.digitaltrends.com/computing/microsoft-wont-include-support-for-silverlight-in-windows-10-edge-browser/>>. 4.7.2015. Luettu 12.9.2017.
- 24 Windows 10 update history. 2017. Verkkodokumentti. Microsoft. <<https://support.microsoft.com/en-us/help/4000823>>. Päivitetty 5.5.2017. Luettu 12.9.2017.
- 25 HTML5 A vocabulary and associated APIs for HTML and XHTML. 2014. Verkkodokumentti. W3C Recommendation. <<https://www.w3.org/TR/html5/>>. 28.10.2014. Luettu 29.8.2017.
- 26 HTML Canvas Tutorial. Verkkodokumentti. W3schools.com. <[https://www.w3schools.com/graphics/canvas\\_intro.asp](https://www.w3schools.com/graphics/canvas_intro.asp)>. Luettu 16.8.2017.
- 27 HTML Canvas Reference. Verkkodokumentti. W3schools.com. <[https://www.w3schools.com/graphics/canvas\\_reference.asp](https://www.w3schools.com/graphics/canvas_reference.asp)>. Luettu 16.8.2017.
- 28 HTML Canvas Reference. Verkkodokumentti. W3schools.com. <[https://www.w3schools.com/tags/ref\\_canvas.asp](https://www.w3schools.com/tags/ref_canvas.asp)>. Luettu 16.8.2017.

- 29 Kelch, Mara. 2016. Gründe für die Umstellung von Flash auf HTML5. Verkkodokumentti. <<https://blog.scireum.de/2016/06/28/gruende-fuer-umstellung-von-flash-auf-html5/>>. 28.6.2016. Luettu 12.9.2017.
- 30 HTML5 Video. Verkkodokumentti. W3schools.com. <[https://www.w3schools.com/html/html5\\_video.asp](https://www.w3schools.com/html/html5_video.asp)>. Luettu 30.8.2017.
- 31 HTML Multimedia. Verkkodokumentti. W3schools.com. <[https://www.w3schools.com/html/html\\_media.asp](https://www.w3schools.com/html/html_media.asp)>. Luettu 30.8.2017.
- 32 The Ogg container format. Verkkodokumentti. Xiph.org. <<https://xiph.org/ogg/>>. Luettu 27.8.2017.
- 33 AVC. Verkkodokumentti. AfterDawn Oy. <<http://fin.afterdawn.com/sanasto/selitys.cfm/avc>>. Luettu 27.8.2017.
- 34 Pfeiffer, Silvia & Pearce, Chris. 2011. The Definitive Guide to HTML5 Video. Apress.
- 35 About WebM. Verkkodokumentti. The WebM Project. <<https://www.webmproject.org/about/>>. Luettu 27.8.2017.
- 36 More Data Supports Personalization of Emails. 2014. Verkkodokumentti. Marketingcharts. <<http://www.marketingcharts.com/industries/media-and-entertainment-39688>>. 10.2.2014. Luettu 14.9.2017.
- 37 While Recognizing Its Importance, Most Marketers Say They Struggle With Personalization. 2013. Verkkodokumentti. Marketingcharts. <<http://www.marketingcharts.com/industries/retail-and-e-commerce-28875>>. 22.4.2013. Luettu 14.9.2017.
- 38 Litt, Michael. Video Marketing Metrics: Get the Latest Research Report. Verkkodokumentti. <<https://www.vidyard.com/blog/b2b-video-marketing-metrics-research-report/>>. Luettu 14.9.2017.
- 39 What's new in PDF Print Engine 4. 2017. Verkkodokumentti. Adobe. <<http://www.adobe.com/uk/products/pdfprintengine.html>>. Luettu 6.9.2017.
- 40 HTML5 Canvas. 2013. Verkkodokumentti. O'scon. <<http://chimera.labs.oreilly.com/books/1234000001654/ch06.html>>. Luettu 6.9.2017.
- 41 Hajautettu sisällönjakelu (CDN). Verkkodokumentti. Suomen Hostingpalvelu Oy. <<https://www.hostingpalvelu.fi/hajautettu-sisallonjakelu-cdn/>>. Luettu 2.9.2017.
- 42 Canvas API. Verkkodokumentti. MDN Web Docs. <[https://developer.mozilla.org/en-US/docs/Web/API/Canvas\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API)>. Päivitetty 20.9.2017. Luettu 30.8.2017
- 43 Manipulating video using canvas. Verkkodokumentti. MDN Web Docs. <[https://developer.mozilla.org/en-US/docs/Web/API/Canvas\\_API/Manipulating\\_video\\_using\\_canvas](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Manipulating_video_using_canvas)>. Päivitetty 25.5.2017. Luettu 12.9.2017.

- 44 Ojanen, Kauko. 2013. Televisio- ja verkkovideotekniikka. Kurssimateriaali. Metropolia Ammattikorkeakoulu.
- 45 Wikström, Pekka. 2010. Vaihtuvan tiedon käyttöönotto ja soveltaminen pieneen graafisen alan yritykseen. Insinööriyö. Metropolia Ammattikorkeakoulu. Theseus-tietokanta.
- 46 Tab Atkins Jnr. video + canvas = magic. 2010. Verkkodokumentti. HTML5 Doctor. <<http://html5doctor.com/video-canvas-magic/>>. 20.10.2010. Luettu 30.8.2017.
- 47 Beattie, Steve. Will HTML5 be able to continue its dominance in 2017? Verkkodokumentti. Quora. <<https://www.quora.com/Will-HTML5-be-able-to-continue-its-dominance-in-2017>>. Luettu 17.9.2017.
- 48 Areenan selainpalvelun uusi soitin helpottaa käyttöä. 2017. Verkkodokumentti. Yle. <<https://yle.fi/aihe/artikkeli/2017/06/28/areenan-selainpalvelun-uusi-soitin-helpottaa-kayttoa>>. Päivitetty 25.10.2017. Luettu 2.10.2017.
- 49 Video Content Marketers Enthusiastic About Channel's Effectiveness. 2014. Verkkodokumentti. Marketingcharts. <<http://www.marketingcharts.com/digital-45556>>. 22.4.2013. Luettu 17.9.2017.
- 50 Sinun tietosi eivät ole sinun – Näin mainostajat keräävät ja hyödyntävät dataasi, kun muutat, ostat kilon pekonia tai haaveilet lomareissusta. 2017. Verkkodokumentti. Yle. <<https://yle.fi/uutiset/3-9901680>>. Päivitetty 30.10.2017. Luettu 29.10.2017.
- 51 WebGL. 2015. Verkkodokumentti. MDN Web Docs. <[https://developer.mozilla.org/fi/docs/Web/API/WebGL\\_API](https://developer.mozilla.org/fi/docs/Web/API/WebGL_API)>. Päivitetty 27.8.2015. Luettu 10.9.2017.
- 52 Canvas (basic support). 2017. Verkkodokumentti. Can I Use... <<http://caniuse.com/#search=canvas>>. Päivitetty 29.10.2017. Luettu 22.8.2017.
- 53 WebM video format. 2017. Verkkodokumentti. Can I Use... <<http://caniuse.com/#search=webm>>. Päivitetty 29.10.2017. Luettu 22.9.2017.
- 54 Ogg Vorbis audio format. 2017. Verkkodokumentti. Can I Use... <<http://caniuse.com/#search=ogg>>. Päivitetty 29.10.2017. Luettu 22.9.2017.
- 55 MPEG-4/H.264 video format. 2017. Verkkodokumentti. Can I Use... <<http://caniuse.com/#search=h.264>>. Päivitetty 29.10.2017. Luettu 22.9.2017.
- 56 HTML 5.2. 2017. Verkkodokumentti. W3C. <<https://www.w3.org/TR/html52/>>. Päivitetty 2.11.2017. Luettu 15.10.2017.
- 57 Usage statistics and market share of HTML5 for websites. 2017. Verkkodokumentti. W3Techs. <<https://w3techs.com/technologies/details/ml-html5/all/all>>. Päivitetty 2.11.2017. Luettu 5.10.2017.
- 58 Mitä on responsiivinen suunnittelu? Verkkodokumentti. Responsiivisuus.info. <<http://www.responsiivisuus.info/responsiivisuus.html>>. Luettu 5.10.2017.

- 59 HTML5:stä rakentuu webin tulevaisuus. 2010. Verkkodokumentti. Micropc.net pdf. Luettu 15.10.2017.
- 60 Krill, Paul. 2017. HTML5: Where the core web technology is headed now. Verkkodokumentti. <<https://www.infoworld.com/article/3198460/web-development/html5-where-the-core-web-technology-is-headed-now.html>>. 26.5.2017. Luettu 15.10.2017.
- 61 Marketers Struggling to Reinvent Themselves in Digital Age, Adobe Study Reveals. 2014. Verkkodokumentti. Adobe. <<http://news.adobe.com/press-release/adobe-marketing-cloud/marketers-struggling-reinvent-themselves-digital-age-adobe>>. 25.3.2014. Luettu 15.10.2017.
- 62 Smith, Brad. 2017. "Hey \$FNAME" is Dead. Here's How to Personalize Marketing in 2017. Verkkodokumentti. <<https://blog.kissmetrics.com/personalize-marketing-in-2017>>. Luettu 15.10.2017.
- 63 Oetting, Jami. 2015. The Next Phase in Personalization? Video Marketing. Verkkodokumentti. <<https://blog.hubspot.com/agency/personalization-video-marketing>>. Päivitetty 2.11.2017. Luettu 15.10.2017.
- 64 Flash & The Future of Interactive Content. Verkkodokumentti. Adobe. <<https://blogs.adobe.com/conversations/2017/07/adobe-flash-update.htm>>. 25.7.2017. Luettu 13.10.2017.
- 65 Search product lifecycle. Verkkodokumentti. Microsoft. <<https://support.microsoft.com/en-us/lifecycle/search/?c2=12905>>. Luettu 25.9.2017.
- 66 HTML5 Canvas Element Guide. 2010. Verkkodokumentti. Webpage FX Inc. <<https://www.webpagefx.com/blog/web-design/canvas-element/>>. 12.10.2010. Luettu 13.10.2017
- 67 Video Manipulation with the Canvas Element. 2011. Verkkodokumentti. Codeproject.com. <<https://www.codeproject.com/Articles/242061/Video-Manipulation-with-the-Canvas-Element>>. 17.9.2011. Luettu 15.10.2017.
- 68 Khoury, Adam. 2013. Canvas Bootcamp 12 – Animation and App Initialization. Verkkodokumentti. Youtube. <<https://www.youtube.com/watch?v=LgTSV2QWoTU>> Katsottu 28.8.2017.

## Testityön koodi

```
<!DOCTYPE HTML>
<html>
<head>
<title>Canvas: HTML5</title>
<meta charset="utf-8">
<body>
    <canvas id="myCanvas" style=border: 1px solid; width="640"
height="360">
    Upgrade Browser
    </canvas>

    //

    
    <video id="video" src="truck.mp4" style="display:
none;"></video>
</body>
<script>
window.onload = canvas;

function canvas()
{
    var myCanvas = document.getElementById("myCanvas");

    if (myCanvas && myCanvas.getContext("2d"))
    {
        var context = myCanvas.getContext("2d");

        var myVideo = document.getElementById("video");

        var myImg = document.getElementById("img");
```

```
var x = 1180
var w = 530
var h = 155

myVideo.play();

setInterval(function() {
  var myCanvas = document.getElementById("myCanvas");
  var context = myCanvas.getContext("2d");
  var myVideo = document.getElementById("video");
  context.drawImage(myVideo, 0, 0, 640, 360, 0, 0, 640,
360);

  x = x - 17

  var myImg = document.getElementById("img");
  context.drawImage(myImg, x, 80, w, h);
}, 60);
}
}
</script>
</head>

</html>
```